

# Modeling Attractiveness and Multiple Clicks in Sponsored Search Results

Dinesh Govindaraj  
Microsoft Bing Ads  
Bangalore  
digovind@microsoft.com

Tao Wang<sup>\*</sup>  
Purdue University  
West Lafayette, IN  
taowang@purdue.edu

S V N Vishwanathan<sup>\*</sup>  
Purdue University  
West Lafayette, IN  
vishy@stat.purdue.edu

## ABSTRACT

Click models are an important tool for leveraging user feedback, and are used by commercial search engines for surfacing relevant search results. However, existing click models are lacking in two aspects. First, they do not share information across search results when computing attractiveness. Second, they assume that users interact with the search results sequentially. Based on our analysis of the click logs of a commercial search engine, we observe that the sequential scan assumption does not always hold, especially for sponsored search results. To overcome the above two limitations, we propose a new click model. Our key insight is that sharing information across search results helps in identifying important words or key-phrases which can then be used to accurately compute attractiveness of a search result. Furthermore, we argue that the click probability of a position as well as its attractiveness changes during a user session and depends on the user's past click experience. Our model seamlessly incorporates the effect of externalities (quality of other search results displayed in response to a user query), user fatigue, as well as pre and post-click relevance of a sponsored search result. We propose an efficient one-pass inference scheme and empirically evaluate the performance of our model via extensive experiments using the click logs of a large commercial search engine.

## Categories and Subject Descriptors

H.4 [Information Systems Applications]: Miscellaneous

## General Terms

Algorithms

## Keywords

Click Models, User behavior, Query log, Click and Browsing data analysis, Sponsored Search, Optimization, Ranking

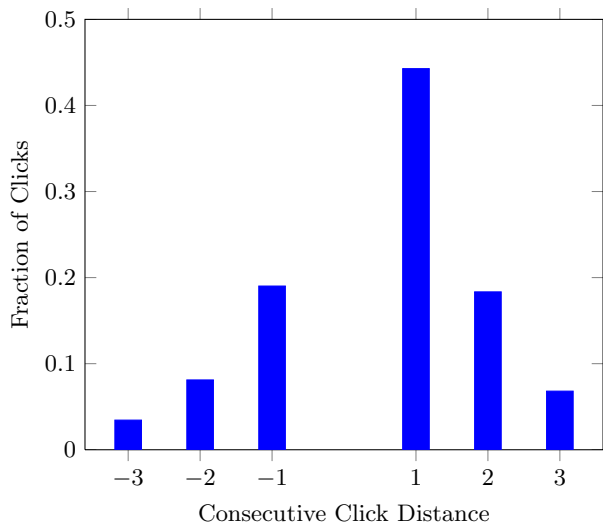
<sup>\*</sup>Work done when visiting Microsoft Bing Ads, Bangalore.

## 1. INTRODUCTION

When a user submits a query, search engines return *organic search results* as a list of ranked URLs. Sometimes URLs linked to landing pages of ads (sponsored search results) are also displayed along with the organic search results. Sponsored search URLs displayed above the organic search results are called *mainline ads*, while those displayed on the side are called *sidebar ads*. Search engines generate revenue when users click on ads. In this paper we are interested in modeling how users interact with and click on sponsored search results. In particular, we focus on mainline ads since a majority of user clicks and hence revenue can be attributed to these ads.

Mining click-through logs is an important component of the never ending quest of commercial search engines to surface the most *relevant* (sponsored) search results in response to an user query. To understand this, consider the following scenario: Suppose documents  $d$  and  $d'$  are displayed in response to a query  $q$ . If the number of clicks that  $d$  receives is disproportionately high compared to  $d'$ , one can reasonably conclude that  $d$  is more relevant than  $d'$  for  $q$ . While this approach of learning from the “wisdom of the crowd” is cheaper than employing a human labeler, it is not without its fair share of difficulties. For instance, it is well known that user clicks display a position bias, that is, documents at higher ranked positions are more likely to be clicked than lower ranked documents. There is a rich body of research which tries to infer an unbiased estimate of the document relevance from click-through logs by explicitly modeling user behavior using click models.

Almost all existing click models are designed for organic search, and make the simplifying assumption that users interact with the search results sequentially [12, 13]. In other words, they assume that a user examines an URL at position  $i + 1$  only after examining the URLs at positions  $1, \dots, i$ . Furthermore, since an overwhelming majority of user sessions end after one click, many models focus exclusively on this scenario. However, based on the analysis of the click logs of a commercial search engine, we observed that user interaction with sponsored search results do not confirm to these assumptions. Approximately 10% of the user sessions with clicks contain more than one click. Furthermore, in approximately 30% of multi-click sessions at least one pair of clicks is in reverse order (*e.g.*, a click at position 3 followed by a click at position 1). This situation is depicted graphically in Figure 1.



**Figure 1:** If the user clicks on position  $i$  followed by position  $j$ , then the click distance is defined as  $j - i$ . Since at most 4 mainline ads are shown, the click distance lies in the range  $[-3, 3]$ . We plot a normalized histogram of the distance between consecutive clicks.

We conjecture that users scan all the sponsored search results before they begin clicking. This is because sponsored search results are shown in a small block at the beginning of the page and usually occupy only 4 to 8 lines. Perhaps, eye tracking studies [9] which are often used to support the linear scan assumption in organic search may not apply to sponsored search<sup>1</sup>. Therefore, meaningful click models for sponsored search need to model reverse clicks.

Unlike organic search, where short snippets of the landing page of a search result is displayed, sponsored search results are designed to grab attention. Towards this end, they employ titles that are short, catchy, and some words (which match query terms) are displayed in bold font. A sponsored search result might look very attractive to an user, but after clicking on it she might realize that the landing page does not contain the information that she wants. In other words, not only the *attractiveness* but also the *post-click relevance* of a sponsored search result are important factors for user satisfaction in sponsored search.

Somewhat surprisingly, most existing click models either do not distinguish between attractiveness and post-click relevance or use very simple methods of estimating attractiveness. However, consider the following: the presence of the words “bonded and insured” in the title can make a sponsored search result more attractive for the query “plumber”. If we can share such information across all sponsored search results displayed in response to the query “plumber”, we can accurately estimate attractiveness.

<sup>1</sup>Validating our conjecture using carefully designed user studies is an interesting research direction in its own right, but is unfortunately beyond the scope of the current work

In this paper, we present a novel statistical click model which models reverse clicks, and incorporates a new method for estimating attractiveness. Like in previous work [15, 16, 5] we also assume a separable model, that is, the user propensity for clicking on a search result is a product of the probability of examining that position and the attractiveness of the document displayed at that position. However, our key insight is that the examination probability of a position as well as the attractiveness of a position changes over time and depends on the user’s past click experience.

Our contributions can be summarized as follows: We propose a new click model for user interaction with sponsored search results. Our model can handle multiple click sessions as well as user sessions where the order of clicks is reversed. Furthermore, we present a new method for estimating attractiveness which shares information across multiple sponsored search results displayed in response to a user query. Our model is flexible and can easily incorporate user fatigue, pre-click and post-click relevance of an ad, and externalities. We derive an efficient one-pass inference mechanism for estimating parameters. Finally, we show that our model comprehensively outperforms existing click models in large scale empirical evaluation using real-life data from a large commercial search engine.

## 2. CLICK MODELS: A BRIEF REVIEW

In what follows we will use the terms documents, ads, urls, and sponsored search results interchangeably. Almost all click prediction models make the following separability assumptions: (1) A document is clicked if and only if it is examined (or viewed); (2) The probability that a document is clicked is independent of its position given that it was viewed; (3) The probability that a document is viewed is independent of the document given the position, and is independent of the other ads presented.

**Examination Hypothesis.** Let  $E_i$  (resp.  $C_i$ ) be a random variable which is equal to 1 if a document  $d_i$  at position  $i$  was examined (resp. clicked). Based on the above assumptions one can write:

$$P(C_i = 1 | d_i) = P(C_i = 1 \cap E_i = 1 | d_i) \\ = \underbrace{P(C_i = 1 | d_i, E_i = 1)}_{\text{relevance}} \underbrace{P(E_i = 1)}_{\text{position bias}}. \quad (1)$$

In other words, the probability of a document being clicked can be factored into the product of the relevance of the document and the position bias. This is the so-called examination hypothesis [15] or separability assumption [1].

**Cascade Model and Extensions.** The cascade model [6] assumes that users scan the documents from top to bottom without skipping, that is, users examine the ads sequentially:

$$P(E_i = 1 | E_{i-1} = 1) = 1 \text{ and } P(E_i = 1 | E_{i-1} = 0) = 0. \quad (2)$$

The cascade model [6] further assumes that users examine documents until they find an appropriate one, and then abandon the search after the first click:

$$P(E_i = 1 | E_{i-1} = 1, C_{i-1}) = 1 - C_{i-1}. \quad (3)$$

The basic cascade model can only deal with query sessions with a single click. The dependent click model (DCM) [11] extends this to multi-click sessions, by modifying (3) to

$$P(E_i = 1 | E_{i-1} = 1, C_{i-1}) = (\lambda_{i-1})^{C_{i-1}}. \quad (4)$$

Here  $\lambda_i$  is estimated from the empirical probability that a browsing session did not end after a click at position  $i$ .

The user browsing model (UBM) [7] extends the DCM by assuming that the examination probability depends on the last clicked position in the same query session. In other words

$$P(E_i = 1 | C_1, \dots, C_i) = \beta_{r_i}, \quad (5)$$

where  $r_i$  denotes the position of the previous click, that is,  $r_i := \operatorname{argmax}_j I(C_j = 1)$ . The Bayesian browsing model (BBM) [14] models relevance  $P(C_i = 1 | d_i, E_i = 1)$  as a random variable, and uses a Bayesian approach to estimate its value.

**Dynamic Bayesian Network Model.** The dynamic Bayesian network (DBN) [4] model differs from the above click models in two important ways. First, it distinguishes between attractiveness (also called perceived relevance or pre-click relevance in literature [16]) and post-click relevance. Second, it incorporates user fatigue. This leads to

$$P(C_{i-1} = 1 | d_{i-1}, E_{i-1} = 1) = \theta_{d_{i-1}} \quad (6)$$

$$P(E_i = 1 | E_{i-1} = 1, C_{i-1}) = \lambda_{i-1} (1 - \rho_{d_{i-1}})^{C_{i-1}}. \quad (7)$$

Here,  $\rho_{d_{i-1}}$  (resp.  $\theta_{d_{i-1}}$ ) denotes the post-click satisfaction (resp. attractiveness) of document  $d_{i-1}$  displayed at position  $i - 1$ . For simplicity, one can set  $\lambda_{i-1} = \lambda$  for all  $i$  [4], or like in the DCM estimate  $\lambda_{i-1}$  from empirical probabilities. Similarly,  $\theta_{d_{i-1}}$  is estimated (essentially) by counting the number of times a document was displayed and the number of times it was clicked. As we will see later in the paper, estimating the attractiveness by using a naive-Bayes like model leads to information sharing and hence better estimates for the attractiveness.

Recently [3] extended the DBN by learning a per-query fatigue parameter and adding a variable to indicate if the user is likely to interact with sponsored search results or directly skip over them. A related model to DBN is the click chain model (CCM) [10].

**Temporal Hidden Click Model.** Some recent work [18] has focused on incorporating revisiting behaviors into click models. However, the temporal hidden click model (THCM) proposed by the authors of [18] does not distinguish between attractiveness and post-click relevance. Furthermore, THCM allows for only two kinds of transitions namely

$$P(E_i = 1 | E_{i-1} = 1) = \alpha \quad (8)$$

$$P(E_{i-2} = 1 | E_{i-1} = 1) = \gamma, \quad (9)$$

with  $\alpha + \gamma \leq 1$ . Note that the transition probabilities do not depend on the location  $i - 1$ . Furthermore, the probability of examining a position decays exponentially with distance from the current location.

**Effect of Externalities.** Externalities [8] refer to the observation that the click on a document also depends on the quality of other documents presented on the same search result page. That is, the examination event will be affected not only by the documents shown above a certain position, but also by the documents below a certain position.

Much of the early work on click models considers the displayed documents as being independent of each other. Some recent work has tried to take externalities into account. For instance, [17] proposed a conditional random fields based model to click prediction to consider the externalities., [19] considers externalities but restrict their attention to sessions with just two ads. On the other hand, [16] showed that the relevance of a document at a position is not constant and it is affected by the clicks in other positions. In the context of news recommendation, [2] extended the basic DBN model to take into account a sub-modular information gain score, which affects the relevance of a document. Although a similar extension is also possible in our setting, for the sake of brevity we will omit discussing this for our model.

### 3. MODEL DESCRIPTION

#### 3.1 Notation

Since we have to deal with reverse clicks, our notation is non-standard. Given a query  $q$ , we assume that the user is presented with a ranked list of  $n$  documents or sponsored search results  $\mathcal{D} = \{d_1, \dots, d_n\}$ , and she might choose to click on  $0 \leq \hat{n} \leq n$  documents. We consider this as a query session. Let  $c_i \in \{0, 1, \dots, n\}$  be a multinomial random variable;  $c_i$  for  $1 \leq i \leq \hat{n}$  denotes the position of the  $i$ -th click, and we let both  $c_0$  and  $c_{\hat{n}+1}$  to be equal to 0. Let  $\mathbf{c}_{1:i} = (c_1, c_2, \dots, c_i)$ , and  $\mathbf{c}$  be a shorthand for  $\mathbf{c}_{1:\hat{n}+1}$ .

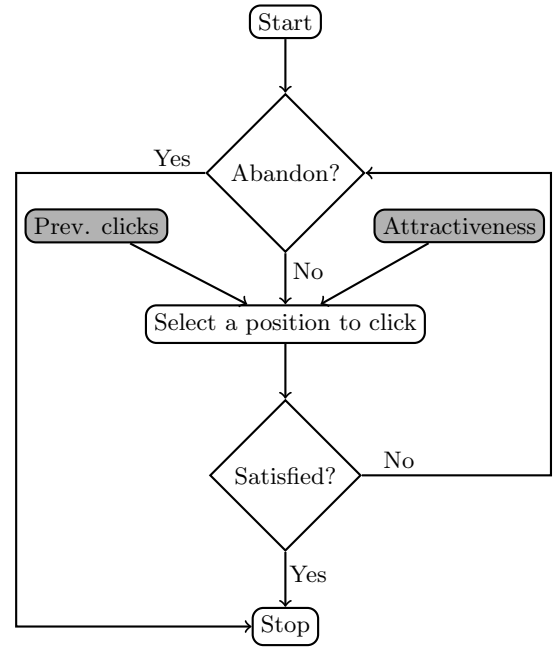


Figure 2: User browsing behavior.

The user browsing behavior in sponsored search can be ex-

plained by Figure 2. After clicking on a document, the user has two choices: she can abandon the session or she can choose to return and click on any other document that has not been clicked so far. Users abandon a session for one of two reasons:

- In the *abandon with satisfaction* case, the user found the information she was looking for in the sponsored search results. We capture this behavior via a Bernoulli random variable  $s_i$ , which takes on a value of 1 when the user is satisfied after clicking the document  $d_{c_{i-1}}$  at position  $c_{i-1}$ . This, in turn, depends on the *post-click relevance* of the clicked document, which we denote as  $\rho_{d_{c_{i-1}}}$ . In other words,

$$P(s_i = 1 | c_{i-1}, \mathcal{D}) = \rho_{d_{c_{i-1}}}. \quad (10)$$

By default, we will set  $P(s_i = 1 | c_0, \mathcal{D}) = 0$ .

- In the *abandon without satisfaction* case, the user may choose to skip ahead to the organic search results or close the browsing session. This behavior is captured via another Bernoulli random variable  $t_i$ , which depends upon the *perseverance* of the user. We will let

$$P(t_i = 1 | s_i = 0, c_{i-1}) = \eta_i, \quad (11)$$

where  $\eta_i$  denotes the probability that user will abandon the session after  $i$  clicks.

As one can expect, the perseverance of the user decreases as the number of clicks increase. This allows our model to capture the effect of externalities in the following natural way: The user fatigue parameter multiplied by the pre-click relevance gives the instantaneous pre-click relevance of a document. If an ad appears alongside other highly relevant ads which have already received clicks, then its instantaneous pre-click relevance reduces. In other words, the number of previous clicks observed is a direct measure of *externalities* (relevance of other documents). On the other hand, when the user decides to continue, then two factors influence the position of the next click:

- She will choose a position based on *attractiveness* of the documents presented. We denote the attractiveness of the document at position  $c_i$  by  $a_{c_i}$ , and assume that the attractiveness of the documents at different positions is independent of each other. Furthermore, let  $\mathbf{a}$  denote the  $n$ -dimensional vector of  $a_i$  and

$$P(a_{c_i} = 1 | \mathcal{D}) = \theta_{d_{c_i}}. \quad (12)$$

- The position of the previous click influences the position of the next click, the so-called *position bias* effect. This is captured using  $\mathbf{v}_i$  a  $n$  dimensional random vector, whose distribution is given by  $\gamma_{c_{i-1}}$

$$P(v_{i,j} = 1 | c_{1:i-1}) = \gamma_{c_{i-1},j}. \quad (13)$$

Here  $\gamma_{c_{i-1},j}$  denotes the probability that the user transitions to position  $j$  after the previous click at position  $c_{i-1}$ . We allow for non-zero  $\gamma_{c_{i-1},j}$  even if  $j < c_{i-1}$ . This is different from existing click models which assume that users scan the list linearly, that is  $c_i > c_{i-1}$ .

### 3.2 Graphical Model

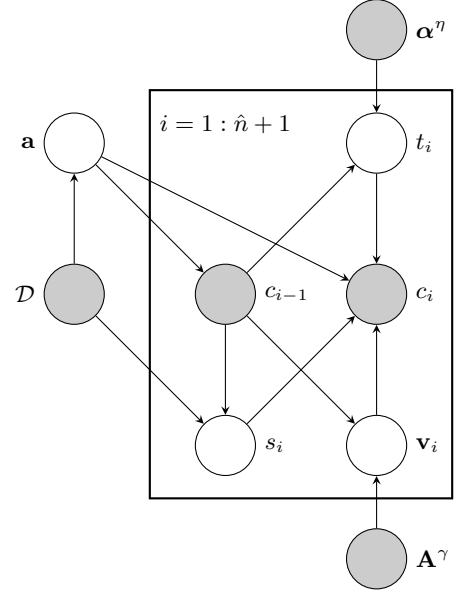
Following standard practice, we will place a *Beta* prior on  $\eta_i$

$$\eta_i \sim \text{Beta}(\alpha_i^\eta, \beta_i^\eta) \quad (14)$$

and a Dirichlet prior on  $\gamma_i$

$$\gamma_i \sim \text{Dirichlet}(\alpha_i^\gamma). \quad (15)$$

A Bayesian network specification of our model can be found in Figure 3. We denote all vectors in bold and matrices in



**Figure 3: Our click prediction model represented as a Bayesian Network. Shaded nodes are observed.**

capital and bold. If we let  $\mathbf{s} = (s_1, \dots, s_n)$ ,  $\mathbf{t} = (t_1, \dots, t_n)$ ,  $\mathbf{V} = (\mathbf{v}_1, \dots, \mathbf{v}_n)$ ,  $\alpha^\eta = ((\alpha_1^\eta, \beta_1^\eta), \dots, (\alpha_n^\eta, \beta_n^\eta))$ , and  $\mathbf{A}^\gamma = (\alpha_1^\gamma, \dots, \alpha_n^\gamma)$ , then our click model can be described as follows:

$$P(\mathbf{c}, \mathbf{s}, \mathbf{t}, \mathbf{a}, \mathbf{V} | \alpha^\eta, \mathbf{A}^\gamma, \mathcal{D}) = \quad (16)$$

$$P(\mathbf{a} | \mathcal{D}) \times P(\mathbf{s} | \mathcal{D}) \times P(\alpha^\eta | \mathcal{D}) \times P(\mathbf{A}^\gamma | \mathcal{D}) \times$$

$$\prod_{i=1}^{\hat{n}+1} P(c_i | s_i, t_i, \mathbf{a}, \mathbf{v}_i) \times P(t_i | c_{i-1}, \alpha^\eta) \times$$

$$P(\mathbf{v}_i | c_{i-1}, \mathbf{A}^\gamma) \times P(s_i | c_{i-1}, \mathcal{D}),$$

where we define

$$P(c_i = 0 | s_i = 1) = 1 \quad (17a)$$

$$P(c_i = 0 | t_i = 1) = 1 \quad (17b)$$

$$P(c_i = j | s_i, t_i = 0, \mathbf{a}, \mathbf{v}_i) = P(a_j = 1) \cdot P(v_{i,j} = 1). \quad (17c)$$

If the user has abandoned the session ( $s_i = 1$  or  $t_i = 1$ ) then subsequent value of  $c_i$  is set to 0. This is captured in (17a) and (17b) above. On the other hand, if the user decides to continue, then the probability of a click at the  $j$ -th position (with  $j \neq 0$ ) depends on the attractiveness of the document as well as the position bias as can be seen from (17c).

### 3.3 Inference

Global parameters of our model include  $\eta_i$ , and  $\gamma_i$  for  $i = 1, \dots, n$ . Furthermore, for each document  $d$  we need to infer its attractiveness  $\theta_d$  and post-click relevance  $\rho_d$ . The hyper-parameters  $\alpha^\eta$  and  $\mathbf{A}^\gamma$  are estimated from historical data, and the other parameters such as  $\eta_i$ ,  $\gamma_i$ , and  $\rho_d$  can be estimated efficiently by one pass through the click-logs. Details can be found in Appendix A. Next we discuss a novel method for computing  $\theta_d$ .

### 3.4 Estimating Attractiveness

A sponsored search result  $d$  displayed in response to a query  $q$  contains three components namely a title, description, and a display URL. We use the following naive Bayes like scheme to estimate attractiveness: For each word  $w$ , excluding commonly occurring stop words, which appears either in the title, description or the display URL<sup>2</sup> we compute  $N_w$  the number of occurrences of  $w$  in  $\mathcal{D}_q$ , and  $\hat{N}_w$  the number of occurrences of  $w$  in the subset of  $\mathcal{D}_q$  that received a click. For infrequent queries (search volume less than 50 in a one week window) we let  $\mathcal{D}_q$  be the entire training corpus, while for frequent queries (which occur with frequency greater than 50) we restrict  $\mathcal{D}_q$  to the documents displayed in response to  $q$ . Let  $|d|$  denote the number of words in  $d$  and estimate

$$\theta_d = \frac{1}{|d|} \sum_{w \in d} \frac{\hat{N}_w}{N_w}. \quad (18)$$

In other words, for each word  $w \in d$  we estimate the fraction of times it occurred in a clicked document for query  $q$ , sum the contribution due to each word independently and normalize by the length of  $d$  to obtain  $\theta_d$ . The advantage of our method is that we share information across documents in the collection  $\mathcal{D}_q$ . For instance, if a word  $w$  occurs frequently in documents which are clicked in response to a query, then it is very likely to be a relevant word for that query and our method gives it a high weight. Another advantage of our method is that we are able to address the cold start problem; we can estimate the attractiveness of a new sponsored search result based on the words that appear in the ad copy.

## 4. EXPERIMENTAL EVALUATION

We collected three weeks of click logs from a large commercial search engine, and used the first week data for computing priors, the second week data for training, and third week data for testing. We filter the data and only retain sessions where at least one mainline ad was displayed. Our test setup is closer to a real-world deployment scenario, and somewhat different from the commonly used practice of randomly splitting available data into a training and test set. In particular, we retain all queries in the test set, even if they have not been observed in the training set. In this case, our estimation is purely based on priors, which are computed from the first week of data. We set  $\alpha_i^\eta + \beta_i^\eta = 10$  and  $\alpha_i^\gamma + \beta_i^\gamma = 10$ . Table 1 summarizes our training data statistics. Note that for high decile ( $>1000$ ) queries, there are significantly fewer three and four click sessions than for tail queries.

### 4.1 Baselines

For our experiments we will compare the performance of our algorithm against the following four baselines:

<sup>2</sup>For simplicity the display URL is treated as a single word.

**Table 1: Because of confidentiality concerns we cannot report the size of our data set. Instead, we report data statistics which are normalized by the number of sessions in the 0-10 bucket. For each decile we also show the fraction of sessions which received one to four clicks.**

Freq.	Frac.	1 Click	2 Click	3 Click	4 Click
0-10	1.0	91.6%	7.3%	1%	0.1%
10-50	0.39	91.5%	7.3%	1.1%	0.1%
50-100	0.16	92.3%	6.6%	1%	0.1%
100-500	0.37	93.5%	5.7%	0.7%	0.1%
500-1000	0.15	95.4%	4.1%	0.42%	0.08%
$>1000$	1.0	97.94%	1.9%	0.15%	0.01%

**Table 2: First click distribution across the four mainline positions.**

1	2	3	4
0.708	0.163	0.0787	0.0503

1. Dynamic Bayesian Network (DBN) model: The user perseverance parameter in DBN (see (7)) was set to  $\lambda_{i-1} = \lambda = 0.01$  for all  $i$ . This was found via cross-validation, based on the recommendation of [4].
2. Independent Click Model (ICM): Since the cascade model [6] can only handle single click sessions, one can extend it by setting the user perseverance parameter  $\lambda$  in (4) to one [11]. This yields the ICM.
3. Position Model (PM): In this model, the click sequence only depends on the position of an ad, and is independent of its contents or attractiveness. In other words, the first click is at position one, second click on position two and so on. Because of position bias (see Table 2), this is a strong baseline.
4. Attractiveness model (AM): The click sequence only depends on the attractiveness of an ad (section 3.4), and is independent of its position. In other words, the most attractive ad is clicked first and so on.

### 4.2 Evaluation Plan

In the first part of our experimental evaluation we are mainly interested in understanding how well our algorithm is able to infer post-click relevance. After all, this is the main motivation behind studying click models. Since other models do not distinguish between attractiveness and post-click relevance, we will only compare our algorithm with the dynamic Bayesian network (DBN) model of [4]. Details of this experiment can be found in Section 4.3.

For our second set of experiments (Sections 4.4 to 4.8) we focus on verifying that our model is able to predict the sequence of user clicks accurately. Click models are usually evaluated by computing average perplexity, or the closely related log-likelihood on the test set. Recall that the perplexity for a single user session is computed as

$$p = 2^{-\frac{1}{n} \sum_{i=1}^n C_i \log q_i + (1-C_i) \log(1-q_i)}, \quad (19)$$

where  $q_i$  is the probability of observing a click at position  $i$  as predicted by the model and  $C_i$  indicates if an actual user click was observed at that position.

However, we believe that perplexity alone does not tell the full story. For instance, the perplexity scores of our model (1.1932) and DBN (1.1984) are very similar, but they differ vastly in terms of how well they predict a sequence of user clicks. Similarly, a model which simply predicts  $q_i$  as the empirical click-through rate, especially for high decile queries, achieves very low perplexity but is unable to explain a sequence of user clicks (also see the results for the AM below). Therefore, we adapt a stronger evaluation criterion which is designed to answer the following natural questions:

- How well does the model predict the first click position?
- How well does the model predict two, three, and all four click sequences? This is a very stringent evaluation criterion for multiple click models, and the model wins only if it predicts all clicks in the sequence correctly.
- Even if the model makes mistakes in predicting the actual click sequence, we want to understand whether the actual click sequence ranks high in terms of log-likelihood.
- Does the model predict the top two and three clicks correctly? In other words, the predicted click sequence need not be in the same order as the actual click sequence but the model needs to accurately predict which ads were clicked. For instance, if the actual click sequence was  $\{1, 2\}$  and the model predicts  $\{2, 1\}$  then this is considered to be a correct prediction as per this metric. Note that predicting the top four clicks in a four click session will always be 100% accurate.
- How does the model fare in terms of predicting and ranking of sessions with reverse clicks?

Intuitively, predicting well on higher decile queries is easier than lower decile queries. In order to understand the strengths and weaknesses of various model, for all the above cases we report their performance across different query deciles.

### 4.3 Predicting Post-Click Relevance

We have access to approximately 10,000 (query, ad) pairs from the test dataset which have been labeled as relevant or irrelevant by trained human editors. Note that the human editors look into the landing pages to determine the labels. Following [4] we rank these  $(q, d)$  pairs using a score which is calculated as  $\theta_d \times \rho_d$  (attractiveness times post-click relevance). The documents are ranked based on the computed score, and we measure precision vs recall. The results for our model and DBN are plotted in Figure 4. Clearly our proposed model outperforms DBN consistently across recalls. In particular note that our model has high precision at low recall and is therefore able to rank relevant documents higher on the list as compared to DBN. Our model achieves an AUC of 0.8653 compared to DBN which is only able to achieve 0.7843.

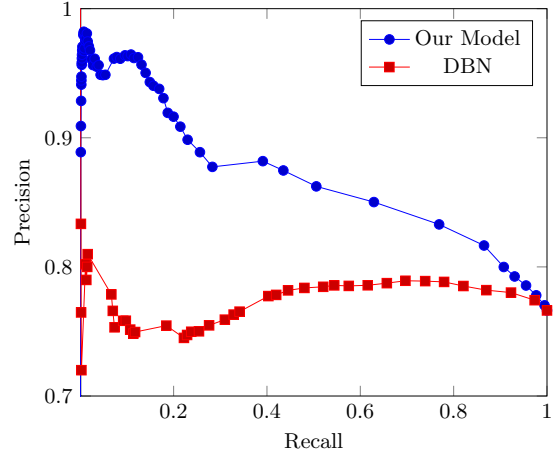


Figure 4: Area Under Curve for predicting relevance of Ads

### 4.4 Predicting the First Clicked Ad

This is a multi-class classification problem with imbalanced class probabilities. Therefore PM which predicts that the first click happens at position one and has an accuracy of 72.27% on our test dataset. The AM has an accuracy of 73.5% while the DBN and ICM accuracies are 72.7% and 71.7% respectively. In our model we predict the ad which has the maximum click propensity as the first clicked ad. This results in an accuracy of 79.59%, a gain of over 8.2% as compared to the other models. To further understand the performance of models across different query deciles we plot the accuracy of different models in Figure 5. Note that we consistently outperform all other models across all query deciles, with the gains being more substantial for the lower decile queries which are harder to learn.

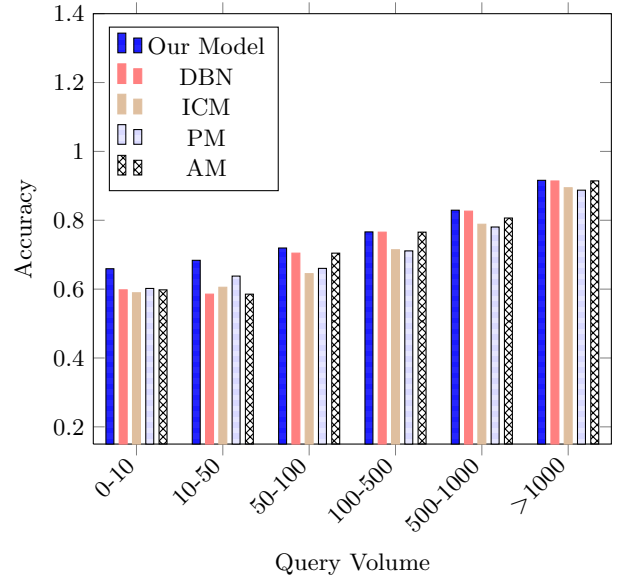


Figure 5: Accuracy of first click sequence prediction across different query deciles.

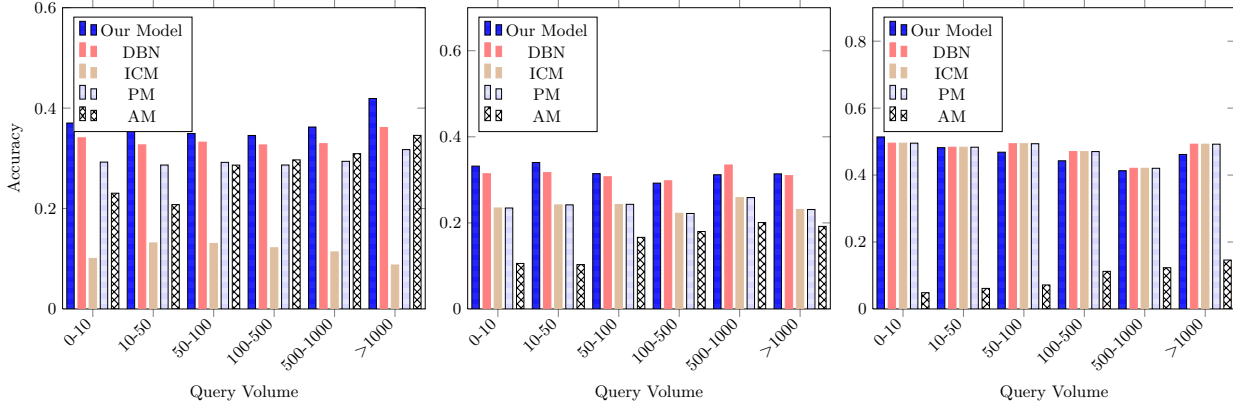


Figure 6: Accuracy of two (L), three (M), four (R) click sequence prediction across different query deciles.

#### 4.5 Predicting the Entire Click Sequence

We compute click propensity for each click sequence with the same length as the actual click sequence. We say that our model predicted the click sequence only when we predict the entire click sequence correctly. Table 3 summarizes the results. Our model gains 8.6% (resp. 25%) on two-click-sequence prediction and 3.4% (resp. 37%) on three-click-sequence prediction over DBN (resp. PM). The model accuracies are comparable when predicting four click sequences, which are only a very small fraction of the data.

Although AM is very competitive when predicting the first click, it is unable to predict longer click sequences accurately. This is because users do not decide to click on an ad solely based on its attractiveness, position bias and past click experience also plays an important role and needs to be taken into account.

Table 3: Accuracy of Predicting the Entire Click Sequence.

# Clicks	Our Model	DBN	PM	AM	ICM
2	36.71	33.79	29.35	25.44	10.94
3	32.26	31.27	23.54	12.80	23.55
4	47.94	48.58	48.58	6.86	48.58

Figure 6 shows the accuracy of the models for two, three and four click prediction across different query deciles. Note that we consistently outperform other models across all query deciles in two click sequence prediction. In three click sequence prediction, DBN performs slightly better in top deciles than our model. Since number of queries and session in tail deciles are higher than top deciles, we achieve better overall performance than DBN.

#### 4.6 Ranking of the Actual Click Sequence

In the previous section we focused on predicting the entire click sequence correctly. Here we focus on how we rank the actual click sequences in terms of log-likelihood. For this, we compute the log likelihood for all permutations of click sequences. We then sort the click sequences based on the computed log likelihood and check the rank of the actual click sequence in this list. Table 4 summarizes our results. As can be seen, our model on the average ranks the actual

Table 4: Overall Actual Click Sequence Ranking.

Number of Clicks	Average Rank
1	$1.25 \pm 0.62$
2	$2.25 \pm 2.27$
3	$2.67 \pm 4.17$
4	$1.87 \pm 3.36$

click sequence among the top 3 or 4 of all the possible permutations of click sequence.

Figure 7 shows how the ranking of the actual click sequence varies across different query deciles. As can be seen from Table 1, longer click sequences are more frequent in the lower deciles, and hence our model has more data to learn in these deciles. Consequently the average rank of the actual click sequence as predicted by our model for two, three and four clicks is lower for lower decile queries.

#### 4.7 Predicting Top Clicks

Here we ignore the order and focus on understanding if our model is able to predict the *positions* of the clicks in two and three click sessions. Table 5 shows we out perform all other models in predicting top 2 and 3 click positions. Figure 8 shows accuracies across query deciles.

Table 5: Accuracy of Predicting the Locations of Top Clicks.

# Clicks	Our Model	DBN	PM	AM	ICM
2	54.93	51.11	43.89	48.5	16.07
3	62.46	59.14	44.98	59.7	45.01

#### 4.8 Predicting Reverse Click Sequences

In our final experiment we focus only on the sessions where at least one pair of clicks was observed in reverse order (reverse click sessions). As before, we focus on the accuracy (Table 6), rank of the actual click sequence (Table 7), and accuracy of predicting the location of the top clicks (Table 8). Figure 9 shows the accuracy and rank of the actual click sequence for different deciles, and Figure 10 shows the accuracy of predicting positions of clicks in two and three click sessions.

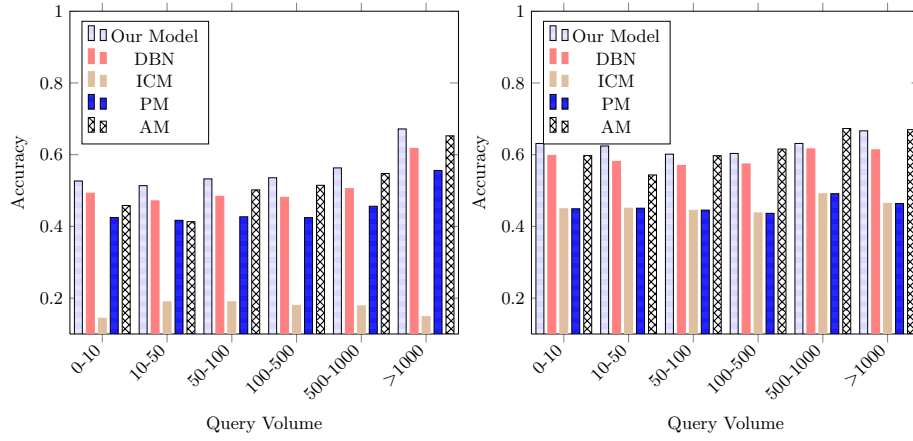


Figure 8: Accuracy of predicting positions of two (L) and three (R) click sessions across different query deciles.

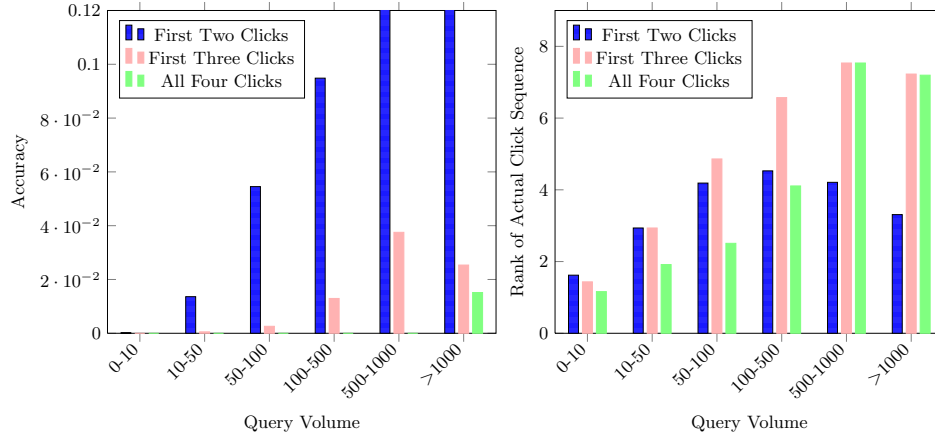


Figure 9: Accuracy (L) of predicting the entire click sequence, and ranking (R) of the actual click sequence across different query deciles for sessions where at least one pair of clicks was observed in reverse order.

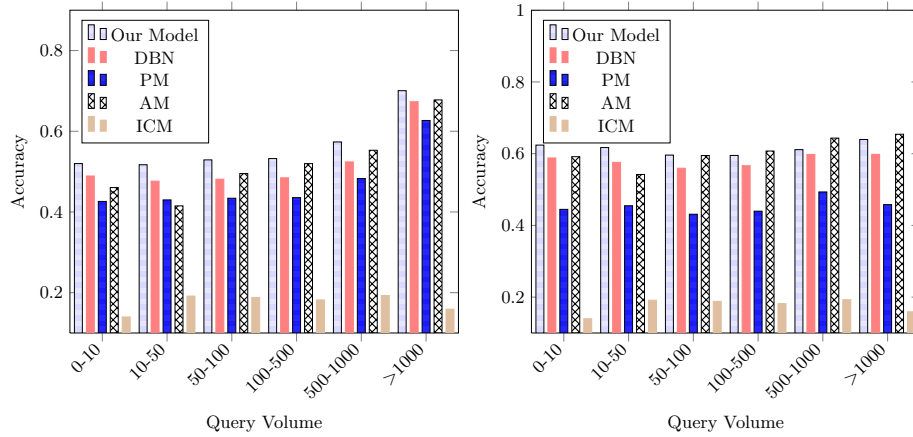


Figure 10: Accuracy of predicting positions of two (L) and three (R) click sessions across different query deciles for sessions where at least one pair of clicks was observed in reverse order.



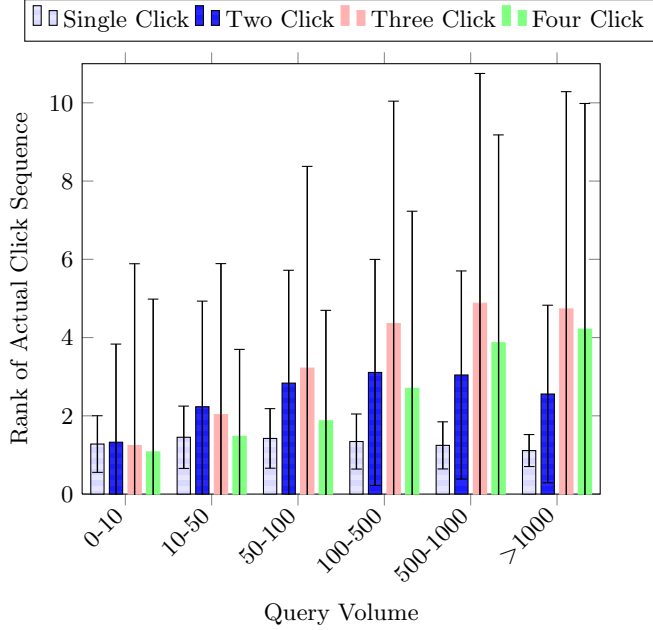


Figure 7: Actual click sequence ranking across different query deciles.

Table 6: Accuracy of Predicting the Entire Click Sequence when Actual Click Sequence has Reverse Order.

# Clicks	Our Model	DBN	PM	AM
2	8.1	0.0	0.0	17.70
3	0.6	0.0	0.0	9.4
4	0.1	0.0	0.0	3.9

As expected our model prediction accuracy for reverse click sessions is low since the reverse click sessions are only around 30% of the multi-click sessions. Even though our accuracies are lower than the attractiveness model, our model ranks actual reverse click sequences in top 5 of all the possible permutations of click sequence. Our model outperforms all other models in most of the experiments and still achieves better ranking of actual click sequence even when clicks have been observed in reverse order.

## 5. CONCLUSION AND FUTURE WORK

We presented a new multiple click model for modeling how users interact with and click on sponsored search results. Our model can handle reverse click sequence and comprehensively outperforms other models across a number of different metrics in extensive empirical evaluation.

Table 7: Overall Reverse Click Sequence Ranking.

Number of Clicks	Average Rank
2	3.15±3.01
3	3.89±5.46
4	2.67±4.56

Table 8: Accuracy of Predicting Locations of Top Clicks when Actual Click Sequence has Reverse Order.

# Clicks	Our Model	DBN	PM	AM	ICM
2	55.74	51.12	45.66	49.50	16.09
3	61.51	58.20	44.74	59.14	44.75

Online sponsored search auctions are priced using a Generalized Second Price (GSP) auction mechanism. Inherent in this model is the assumption that the clicks on the ads happen independent of each other. We are currently working on developing pricing mechanisms which will take into account the clicking behavior predicted by our model. Our efforts are also directed towards improving the reverse click prediction accuracy of our model by using stronger priors. Finally, we are also working towards making our model robust to noise.

## APPENDIX

### A. PARAMETER ESTIMATION

Our training data consists of  $m$  sessions, and we assume that the document set  $\mathcal{D}^k$  was displayed in the  $k$ -th session in response to query  $q^k$  and we observed a click sequence  $\mathbf{c}^k$  of length  $\hat{n}^k$ . If we assume that the sessions are iid, then

$$P(\{\mathbf{c}^k\} | \{\mathcal{D}^k\}) = \prod_{k=1}^m P(\mathbf{c}^k | \mathcal{D}^k). \quad (20)$$

Let  $c_i^k$  denote the location of the  $i$ -th click in the  $k$ -th session,  $I(\cdot)$  denote the indicator variable of an event,  $\mathcal{D}'$  denote the set of unique query-document pairs in our session data and  $D = |\mathcal{D}'|$ . Plugging in (16) into (20), using (10)–(17) shows that  $P(\{\mathbf{c}^k\} | \{\mathcal{D}^k\})$

$$\propto \prod_{d \in \mathcal{D}'} \theta_d^{\psi_d} \times \prod_{j=1}^n \prod_{k=1}^n \gamma_{jk}^{\delta_{jk}} p(\gamma_{jk}) \\ \times \prod_{j=0}^n (1 - \eta_j)^{\beta_j} \eta_j^{\beta'_j} p(\eta_j) \times \prod_{d \in \mathcal{D}'} (1 - \rho_d)^{\kappa_d} \rho_d^{\kappa'_d},$$

where we define

$$\psi_d = \sum_{k=1}^m \sum_{i=1}^{\hat{n}_k} I(d_{c_i^k} = d) \quad (21)$$

$$\delta_{i,j} = \sum_{k=1}^m \sum_{t=1}^{\hat{n}_k} I(c_{t-1}^k = i) \text{ and } I(c_t^k = j). \quad (22)$$

$\psi_d$  counts the number of times document  $d$  occurs in  $\{\mathcal{D}_k\}$ ,  $\delta_{i,j}$  counts the number of times we observed a click at position  $j$  after having observed a click at position  $i$ . Note that when  $i < j$   $\delta_{i,j}$  counts the in-sequence users clicks, and for all  $i > j$ ,  $\delta_{i,j}$  captures out of sequence clicks.

$$\beta_j = \sum_{k=1}^m I(\hat{n}^k > j) \text{ and } \beta'_j = \sum_{k=1}^m I(\hat{n}^k = j) \quad (23)$$

$$(24)$$

$\beta_j$  counts the number of sessions with greater than  $j$  clicks, while  $\beta'_j$  counts the number of sessions with exactly  $j$  clicks. It is easy to see that  $\beta_j = \sum_{j' > j} \beta'_{j'}$ , for  $j \neq 0$ .

$$\kappa_d = \sum_{k=1}^m I(d_{c_{\hat{n}k}} \neq d) \text{ and } I(d_{c_i^k} = d) \text{ for some } i, \quad (25)$$

$$\text{and } \kappa'_d = \sum_{k=1}^m I(d_{c_{\hat{n}k}} = d). \quad (26)$$

$\kappa'_d$  counts the number of sessions which ended after clicking on the document  $d$ , while  $\kappa_d$  is the number of sessions which did not end after a click on document  $d$ .

As a consequence of using the Beta prior, we can estimate

$$\eta_j = \frac{\beta'_j + \alpha_j^\eta}{\beta'_j + \alpha_j^\eta + \beta_j + \beta_j^\eta} \text{ and } \rho_d = \frac{\kappa'_d}{\kappa'_d + \kappa_d} \quad (27)$$

Furthermore, since  $\gamma_i \sim \text{Dir}(\alpha_i^\gamma)$ , we can estimate:

$$\gamma_{i,j} = \frac{\delta_{i,j} + \alpha_{i,j}^\gamma}{\sum_j (\delta_{i,j} + \alpha_{i,j}^\gamma)}. \quad (28)$$

## A.1 Time Complexity

In order to perform the updates (27) and (28) we need to compute the quantities defined in (21) – (26). However, all these quantities involve simple counts, which can be computed by using one-pass through the click logs. Therefore, we can conclude that inference in our model is extremely salable.

## B. REFERENCES

- [1] Gagan Aggarwal, Ashish Goel, and Rajeev Motwani. Truthful auctions for pricing search keywords. In *Proceedings of the 7th ACM conference on Electronic commerce*, pages 1–7, 2006.
- [2] Amr Ahmed, A. J. Smola, Choon Hui Teo, and S. V. N. Vishwanathan. Fair and balanced: Learning to present news stories. In *Web Science and Data Mining (WSDM)*, 2012.
- [3] Azin Ashkan and Charles L.A. Clarke. Modeling browsing behavior for click analysis in sponsored search. In *Proceedings of the 21st ACM Conference on Information and Knowledge Management*, pages 2015–2019, 2012.
- [4] Olivier Chapelle and Ya Zhang. A dynamic bayesian network click model for web search ranking. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pages 1–10, 2009.
- [5] Ye Chen and Tak W. Yan. Position-normalized click prediction in search advertising. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '12, pages 795–803, 2012.
- [6] Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. An experimental comparison of click position-bias models. In *Proceedings of the international conference on Web search and web data mining*, WSDM '08, pages 87–94, 2008.
- [7] Georges E. Dupret and Benjamin Piwowarski. A user browsing model to predict search engine click data from past observations. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 331–338, 2008.
- [8] Arpita Ghosh and Mohammad Mahdian. Externalities in online advertising. In *Proceedings of the 17th international conference on World Wide Web*, WWW '08, pages 161–168, 2008.
- [9] Laura A. Granka, Thorsten Joachims, and Geri Gay. Eye-tracking analysis of user behavior in www search. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pages 478–479, 2004.
- [10] Fan Guo, Chao Liu, Anitha Kannan, Tom Minka, Michael Taylor, Yi-Min Wang, and Christos Faloutsos. Click chain model in web search. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 11–20, 2009.
- [11] Fan Guo, Chao Liu, and Yi Min Wang. Efficient multiple-click models in web search. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, pages 124–131, 2009.
- [12] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD)*. ACM, 2002.
- [13] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, and Geri Gay. Accurately interpreting clickthrough data as implicit feedback. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '05, pages 154–161, 2005.
- [14] Chao Liu, Fan Guo, and Christos Faloutsos. BBM: bayesian browsing model from petabyte-scale data. In *Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 537–546, 2009.
- [15] Matthew Richardson, Ewa Dominowska, and Robert Ragno. Predicting clicks: Estimating the click-through rate for new ads. In *World Wide Web Conference*, 2007.
- [16] Ramakrishnan Srikant, Sugato Basu, Ni Wang, and Daryl Pregibon. User browsing models: relevance versus examination. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '10, pages 223–232, 2010.
- [17] Chenyan Xiong, Taifeng Wang, Wenkui Ding, Yidong Shen, and Tie-Yan Liu. Relational click prediction for sponsored search. In *Proceedings of the fifth ACM international conference on Web search and data mining*, WSDM '12, pages 493–502, 2012.
- [18] Danqing Xu, Yiqun Liu, Min Zhang, Shaoping Ma, and Liyun Ru. Incorporating revisiting behaviors into click models. In *Proceedings of the fifth ACM international conference on Web search and data mining*, WSDM '12, pages 303–312, New York, NY, USA, 2012. ACM.
- [19] Wanhong Xu, Eren Manavoglu, and Erick Cantu-Paz. Temporal click model for sponsored search. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 106–113, 2010.